Investigating streamlining in bacterial genomes (Figure 1)

Ben Temperton

1 Method

1.1 Data Preparation

```
raw.df <- read.delim("../data/raw_img_bacteria.txt.gz",
    header = TRUE)
colnames(raw.df) <- read.table("../data/img.col.names")$V1
# IMG contains some duplicates:
duplicate.ids <- scan("../data/duplicated.species.ids")
raw.df <- raw.df[!(raw.df$img.species.id %in% duplicate.ids),
    ]
raw.df$img.species.id <- as.factor(raw.df$img.species.id)
raw.df <- subset(raw.df, GenomeSize < 1e+07 & GenomeSize >
    1e+05)
raw.df$non.coding <- 100 - raw.df$CodingBaseCountNPPct
```

Now let's remove anything that is a SAG, and anything with more than 35 % noncoding DNA. Then let's label up the points for displaying on the figure.

```
sag.ids <- raw.df[grep("AAA", raw.df$GenomeName), ]$img.species.id
no.sags <- raw.df[!(raw.df$img.species.id %in% sag.ids),
    ]
no.sags <- subset(no.sags, non.coding < 35)
no.sags$type <- "Other"
sar.11.ids <- scan("../data/sar11.ids")
prochlorococcus.ids <- scan("../data/prochlorococcus.ids")
symbiont.ids <- scan("../data/symbiont.ids")
roseobacter.ids <- scan("../data/roseobacter.ids")
vibrio.ids <- scan("../data/vibrio.ids")
alteromonas.ids <- scan("../data/alteromonas.ids")
htcc2255.id <- c(639857003)
OM43.ids <- c(639857020)
bacteroidetes.ids <- scan("../data/bacteroidetes.ids")
```

```
verrucomicrobia.ids <- scan("../data/verrucomicrobia.ids")
no.sags[no.sags$img.species.id %in% sar.11.ids, ]$type <- "SAR11"
no.sags[no.sags$img.species.id %in% prochlorococcus.ids,
    ]$type <- "Prochlorococcus"
no.sags[no.sags$img.species.id %in% htcc2255.id, ]$type <- "Rhodobacteraceae"
no.sags[no.sags$img.species.id %in% symbiont.ids, ]$type <- "Symbiont"
no.sags[no.sags$img.species.id %in% roseobacter.ids,
    ]$type <- "Rhodobacteraceae"
no.sags[no.sags$img.species.id %in% vibrio.ids, ]$type <- "Vibrionaceae"
no.sags[no.sags$img.species.id %in% alteromonas.ids,
    ]$type <- "Alteromonadaceae"
# no.sags[no.sags£img.species.id %in%
# bacteroidetes.ids, ]£type<-'Bacteroidetes'
no.sags[no.sags$img.species.id %in% verrucomicrobia.ids,
    ]$type <- "Verrucomicrobia"
no.sags[no.sags$img.species.id %in% OM43.ids, ]$type <- "OM43"
swan <- read.delim("../data/swan.sigma.counts")
```

Now we can calculate % of non-coding bases in the SAGs from Swan et al.

```
import os
import glob
from Bio import SeqIO
outfile = open('swan.non.coding.pct', 'w')
outfile.write('img.species.id\tnon.coding\n')
for f in glob.glob('./contigs/*.fasta'):
  root, ext = os.path.splitext(os.path.basename(f))
genome_size = 0.0
coding_size = 0.0
with open(f, 'rU') as handle:
for seq_record in SeqIO.parse(handle, 'fasta'):
genome_size += len(seq_record.seq)
with open('./dna/%s.fasta' % root, 'rU') as handle:
for seq_record in SeqIO.parse(handle, 'fasta'):
coding_size += len(seq_record.seq)
outfile.write('%s\t%.4f\n' % (root, 100-(coding_size*100/genome_size)))
outfile.close()
```

Now we can join this to the swan dataset

```
swan.non.coding <- read.delim("../data/swan.non.coding.pct")
swan <- join(swan, swan.non.coding)

## Joining by:  img.species.id
```

```
add_points <- function(data, typeName, color) {
    s <- subset(data, type == typeName)
    points(s$GenomeSize/1e+06, s$non.coding, pch = 16,
        col = color, cex = 1)
}
zones = matrix(c(2, 0, 1, 3), ncol = 2, byrow = TRUE)
layout(zones, widths = c(4/5, 1/5), heights = c(1/5,
    4/5))
background <- subset(no.sags, type == "Other")
x = no.sags$GenomeSize/1e+06
y = no.sags$non.coding
lin.model <- lm(y ˜ x)
xhist = hist(x, plot = FALSE, breaks = 100)
yhist = hist(y, plot = FALSE, breaks = 100)
top = max(c(xhist$counts, yhist$counts))
op <- par(mar = c(3, 3, 1, 1))
plot(background$GenomeSize/1e+06, background$non.coding,
    pch = 16, cex = 0.5, col = "gray")
add_points(no.sags, "Vibrionaceae", "#FF67A4")
add_points(no.sags, "Rhodobacteraceae", "#00C0AF")
add_points(no.sags, "Symbiont", "#E76BF3")
add_points(no.sags, "Alteromonadaceae", "#E58700")
add_points(no.sags, "SAR11", "red")
add_points(no.sags, "Prochlorococcus", "#1FFF66")
add_points(no.sags, "OM43", "#33FFFF")
points(swan$egs/1e+06, swan$non.coding, pch = 16, cex = 1,
    col = "#9900FF")
abline(lin.model)
legend(6.5, 35, c("Vibrionaceae", "Rhodobacteraceae",
    "Symbiont", "Alteromonadaceae", "SAR11", "Prochlorococcus",
    "OM43", "SAG"), col = c("#FF67A4", "#00C0AF", "#E76BF3",
    "#E58700", "red", "#1FFF66", "#33FFFF", "#9900FF"),
    pch = 16, pt.cex = 1.5)
op <- par(mar = c(0, 3, 1, 1))
barplot(xhist$counts, axes = TRUE, ylim = c(0, 200),
    space = 0)
par(mar = c(3, 0, 1, 1))
barplot(yhist$counts, axes = TRUE, xlim = c(0, top),
    space = 0, horiz = TRUE)
par(oma = c(3, 3, 0, 0))
mtext("Genome Length (Mbp)", side = 1, line = 1, outer = TRUE,
    adj = 0, at = 0.8 * (mean(x) - min(x))/(max(x) -
        min(x)))
mtext("% non-coding DNA", side = 2, line = 1, outer = TRUE,
    adj = 0, at = (0.8 * (mean(y) - min(y))/(max(y) -
        min(y))))
```

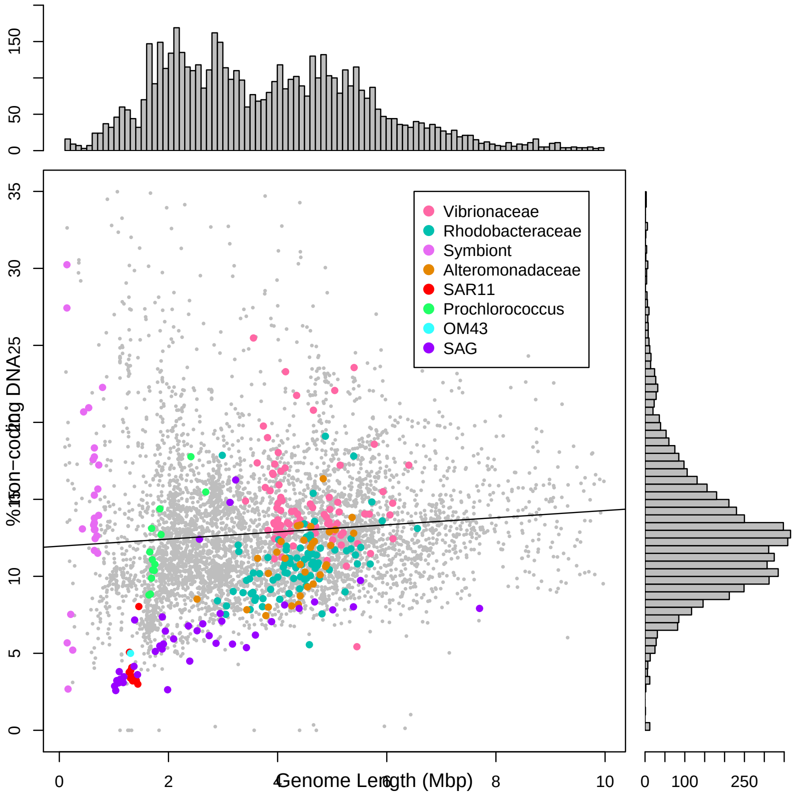

Figure 1: Genome Length vs. pct non-coding DNA

```
par(op)
```

1.2 Is the genome size distribution two populations?

```
library(diptest)
dip.test(no.sags$GenomeSize)
```

```
##
##  Hartigans' dip test for unimodality
##
## data:  no.sags$GenomeSize
## D = 0.0111, p-value = 7.564e-05
## alternative hypothesis: non-unimodal, i.e., at least bimodal
```

Yes, there is evidence of non-unimodality in this population of genome sizes.

1.3 Is there an effect of partially complete genomes?

By using all of the genomes in IMG v400, we are plotting both complete, permanent draft and draft genomes onto Figure 1. Therefore, it is worth testing whether we see a similar pattern when looking only at the genomes with the status of 'Finished'

```
zones = matrix(c(2, 0, 1, 3), ncol = 2, byrow = TRUE)
layout(zones, widths = c(4/5, 1/5), heights = c(1/5,
    4/5))
finished.only <- subset(no.sags, status == "Finished")
finished.background <- subset(finished.only, type ==
    "Other")
x = finished.only$GenomeSize/1e+06
y = finished.only$non.coding
finished.lin.model <- lm(y ~ x)
xhist = hist(x, plot = FALSE, breaks = 100)
yhist = hist(y, plot = FALSE, breaks = 100)
top = max(c(xhist$counts, yhist$counts))
op <- par(mar = c(3, 3, 1, 1))
plot(finished.background$GenomeSize/1e+06, finished.background$non.coding,
    pch = 16, cex = 0.5, col = "gray")
add_points(finished.only, "Vibrionaceae", "#FF67A4")
add_points(finished.only, "Rhodobacteraceae", "#00C0AF")
add_points(finished.only, "Symbiont", "#E76BF3")
add_points(finished.only, "Alteromonadaceae", "#E58700")
add_points(finished.only, "SAR11", "red")
add_points(finished.only, "Prochlorococcus", "#1FFF66")
add_points(finished.only, "OM43", "#33FFFF")
abline(finished.lin.model)
legend(6.5, 35, c("Vibrionaceae", "Rhodobacteraceae",
    "Symbiont", "Alteromonadaceae", "SAR11", "Prochlorococcus",
    "OM43", "SAG"), col = c("#FF67A4", "#00C0AF", "#E76BF3",
    "#E58700", "red", "#1FFF66", "#33FFFF", "#9900FF"),
    pch = 16, pt.cex = 1.5)
op <- par(mar = c(0, 3, 1, 1))
barplot(xhist$counts, axes = TRUE, ylim = c(0, 100),
    space = 0)
par(mar = c(3, 0, 1, 1))
barplot(yhist$counts, axes = TRUE, xlim = c(0, top),
    space = 0, horiz = TRUE)
par(oma = c(3, 3, 0, 0))
mtext("Genome Length (Mbp)", side = 1, line = 1, outer = TRUE,
    adj = 0, at = 0.8 * (mean(x) - min(x))/(max(x) -
        min(x)))
mtext("% non-coding DNA", side = 2, line = 1, outer = TRUE,
    adj = 0, at = (0.8 * (mean(y) - min(y))/(max(y) -
        min(y))))
```

```
par(op)
```

Now, is there still a lack of unimodality?

```
dip.test(finished.only$GenomeSize)
```

```
##
##  Hartigans' dip test for unimodality
##
## data:  finished.only$GenomeSize
## D = 0.0118, p-value = 0.05164
## alternative hypothesis: non-unimodal, i.e., at least bimodal
```

No,there is no evidence for multi-modality in finished genomes only.

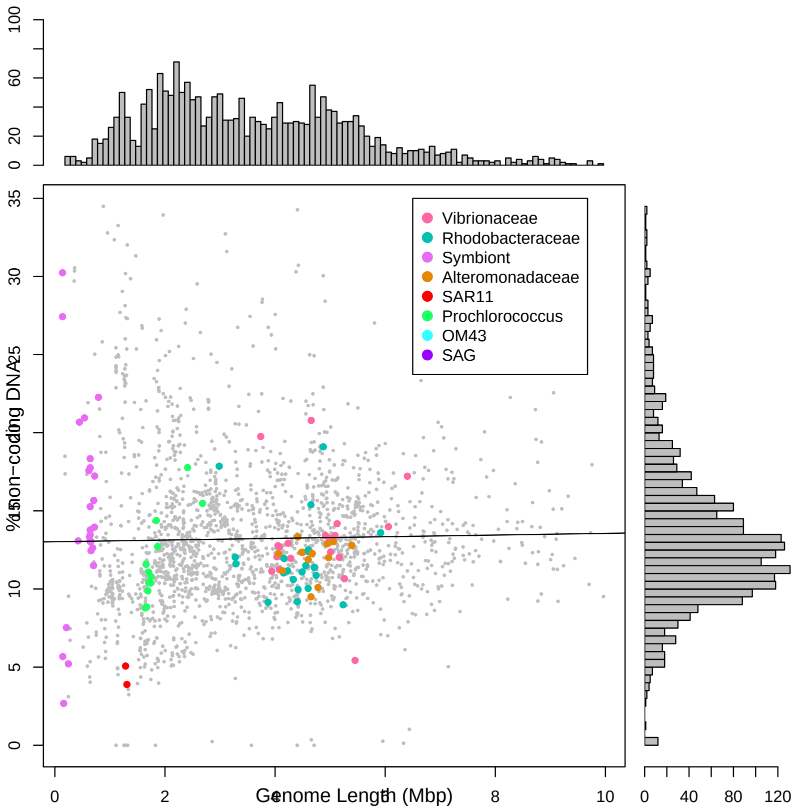

Figure 2: Genome Length vs. pct non-coding DNA for Finished genomes